***Les Cahiers du GERAD*** ISSN: 0711–2440

**Convergence of Variable Neighborhood Search**

J. Brimberg, P. Hansen
N. Mladenović

G–2002–21

May 2002
Revised: January 2004

# Convergence of Variable Neighborhood Search

**Jack Brimberg**
*Department of Business Administration*
*Royal Military College of Canada, Kingston, Canada, K7K 7B4*
jack.brimberg@rmc.ca

**Pierre Hansen**
*GERAD and HEC Montréal*
*3000 Chemin de la Côte-Sainte-Catherine, Montréal, Canada H3T 2A7*
pierre.hansen@gerad.ca

**Nenad Mladenović**
*Mathematical Institute, Serbian Academy of Sciences*
*Belgrade, Yugoslavia*
nenad.mladenovic@gerad.ca

May, 2002
Revised: January, 2004

*Les Cahiers du GERAD*

G–2002–21

**Abstract**

This paper examines the convergence properties of two well-known heuristics: variable neighborhood search (VNS) and random multistart local search (MLS). Both methods are shown to be globally convergent under general conditions. A multistart variable neighborhood search is introduced as a hybrid combination of VNS and MLS. The finite time performance of the three heuristics is compared on a well-known continuous location problem, the multisource Weber problem. This study also attempts to explain why the structured approach of VNS obtains superior results to a multistart local search.

**Keywords:** Variable neighborhood search, Multistart local search, Convergence, Multisource Weber problem

**Résumé**

On étudie les propriétés de deux heuristiques bien connues : la recherche à voisinage variable (RVV) et la recherche locale itérée (RLI). On montre que les deux méthodes sont globalement convergentes sous des hypothèses très générales. Une méthode de recherche à voisinage variable itérée est introduite, comme hybride de RVV et RLI. On compare la performance des trois heuristiques en temps fini pour un problème de localisation bien connu : le problème de Weber multisources. Cette étude s'efforce d'expliquer pourquoi l'approche structurée de RVV donne de meilleurs résultats que RLI.

**Mots-clés :** Recherche à Voisinage Variable, Recherche Locale Itérée, Convergence, Problème de Weber Multisources.

# 1 Introduction

Suppose we have some combinatorial or global optimization problem $\mathcal{P}$ that is formulated as a minimization of a function $f$ on a solution space $S$ that is either a finite set or bounded region. The idea of variable neighborhood search (VNS) is to define a set of neighborhood structures $\mathcal{N}_k$, $k = 1, \ldots, k_{max}$, that can be used in a systematic way to conduct a search through the solution space. Whereas in local search a single neighborhood is typically defined ($k_{max} = 1$), the VNS expands the search over an increasing radius to escape a “local optimum trap”.

The VNS metaheuristic is well-established in the literature. For an overview of the method and numerous applications, the reader is referred to [12], [6], [7]. Despite its popularity, very little effort has been directed towards a theoretical understanding of the method. In this paper we examine a general framework for global convergence of VNS. Some general theory on convergence of metaheuristics exists (e.g., see [10], [9] and [13]); however, we employ a different approach that is based on a concept of “entrapment” probabilities. Through the analysis, the superiority of VNS to random multistart local search (MLS) is also argued. A computational study comparing the finite-time performance of VNS, MLS and a hybrid of the two methods on the continuous multisource Weber problem is given to support our analysis. This study also introduces and examines the concept of entrapment probabilities in different neighborhoods of the incumbent solution, thus presenting a new statistical methodology for investigating the topology of the solution space.

# 2 Review of Basic VNS

To induce a set of neighborhoods $\mathcal{N}_k$ on the solution space $S$, we use a distance function $\rho$ that specifies the distance between any two solutions, $x_1, x_2 \in S$. This may be done for example by comparing the attributes of the two solutions, and setting the distance equal to the number of attributes where $x_1$ and $x_2$ differ; that is, a Hamming distance is defined as

$$\rho(x_1, x_2) = |x_1 \Delta x_2| = |(x_1 \setminus x_2) \cup (x_2 \setminus x_1)|.$$

It is readily shown that $\rho$ is a metric, and $(S, \rho)$ a metric space.

The neighborhood $\mathcal{N}_k(x)$ denotes the set of solutions in the $k^{th}$ neighborhood of $x$, and using the metric $\rho$, is defined as:

$$\mathcal{N}_k(x) = \{y \in S \mid \rho(x, y) = k\}.$$

With this structure applied on the solution space, the basic steps of the VNS metaheuristic are given as follows:

---

*Initialization*. Select the set of neighborhood structures $\mathcal{N}_k$, $k = 1, \ldots, k_{max}$, that will be used in the search; find an initial solution $x$; choose a stopping condition;

*Repeat* the following sequence until the stopping condition is met:

(1) Set $k \leftarrow 1$;

(2) Repeat the following steps until $k = k_{max}$:

*(a) Shaking*. Generate a point $y$ at random from the $k^{th}$ neighborhood of $x$ ($y \in \mathcal{N}_k(x)$);

*(b) Local search*. Apply some local search method with $y$ as initial solution, to obtain a local optimum given by $y'$;

*(c) Move or not*. If this local optimum is better than the incumbent, move there ($x \leftarrow y'$), and continue the search with $\mathcal{N}_1$ ($k \leftarrow 1$); otherwise, set $k \leftarrow k + 1$.

---

The initial solution is usually obtained in VNS by randomly selecting a point in $S$ and descending to a local optimum $x$ using the specified local search. In multistart local search (MLS), a sequence of local optima is obtained by restarting the local search from randomly-selected points in $S$. That is, MLS uses a totally randomized restart mechanism to move to different subregions of S, while VNS uses an anchor point (the incumbent solution) and chooses random points at systematically-varying distances from the anchor point. The parameter $k_{max}$ is typically set as a function of the problem size. For example, in the discrete $p$-median problem, where a given number ($p$) of facilities is to be located at the nodes of a network in order to minimize a sum of transportation costs, we may define the distance between two solutions $x_1$ and $x_2$ simply as the number of facility locations that differ from one solution to the other. In this case, we could set $k_{max} = p$ (or a fraction of $p$). Typically, the neighborhood structure is defined in such a way that when $k$ increases from 1 to $k_{max}$, the size of the neighborhood $\mathcal{N}_k$ (or number of points in $\mathcal{N}_k$) increases at an exponential rate. The more-structured approach of VNS has distinct advantages, as shall be seen.

For comparison purposes, we assume that the MLS and VNS utilize identical local search procedures. However, the local search neighborhood may be defined in any manner and independently of the neighborhood structure imposed on the shaking operation in VNS. This allows a degree of flexibility in the design of both heuristics. In both cases, the best solution is retained after reaching a stopping condition, such as a limit on execution time.

## 3 Global Convergence Properties

Since the two methods we are investigating, VNS and MLS, use a local search, it is useful to consider the subset $L$ of local optima in $S$ obtained by the local search. More precisely, if $\mathcal{L}$ denotes the local search operator, then $\mathcal{L}(x) = x^*$ is a local solution obtained from a series of descent moves using the neighborhood prescribed by the local search. It is assumed that a best-improvement strategy is in effect, and if there are tied solutions in the neighborhood of an intermediate point, any one of the ties is chosen with equal probability; in this latter case the mapping $\mathcal{L}(x)$ may be one-to-many. It follows by definition that $x^*$ is a stationary point of the mapping $\mathcal{L}$, i.e., $\mathcal{L}(x^*) = x^*$.

The set of local solutions may be obtained as:

$$L = \{\mathcal{L}(x) \mid x \in S\}.$$

In combinatorial optimization, $S$, and hence, $L$ may be assumed to contain a finite number of solution points, whereas in continuous space, $S$, and as a result, $L$ may be nondenumerable. Therefore, we take the objective function $f$ that is being minimized, and now consider the set:

$$f(L) = \{f(x) \mid x \in L\}.$$

Setting $L_r = \{x \in L \mid f(x) = f_r\}$, we further assume that $L$ may be divided into a finite number of mutually exclusive subsets, $L_1, \ldots, L_N$ ($\bigcup_{r=1}^{N} L_r = L$), such that all the local optima in $L_r$ have the same objective function value $f_r$, and also order the subsets in decreasing quality of solution so that:

$$f_1 < f_2 < \cdots < f_N,$$

where the global minimum is found in $L_1$.

VNS and MLS are now both conceptualized as generating a finite sequence of incumbent solutions, denoted by $x^q$, $q = 1, \ldots, N'$, where $N' \leq N$. The sequence is monotonic, in the sense that the incumbent solution changes only when an improved local solution is found:

$$f(x^1) > f(x^2) > \cdots > f(x^{N'}).$$

In MLS, a complete iteration occurs from the generation of one local optimum to the next. In VNS, an iteration occurs from one improvement (move) to the next or when the distance $k$ varies through the complete range from 1 to $k_{max}$, whichever comes first. Observe in both methods that the incumbent solution may remain stationary for several successive iterations if no improvement is found.

**Property 1** *Let execution time $t \to \infty$. The resulting finite trajectory $\{x^q; q = 1, 2 \ldots, N'\}$ generated by either MLS or VNS follows a Markov process.*

*Proof.* In MLS the starting point at the beginning of each iteration is chosen at random in the solution space $S$. In VNS the starting points are chosen at random from a predefined sequence of neighborhoods centered at $x^q$ (see the *shaking* operation). It follows in both cases that the transition probabilities to the subsets $L_r$ are independent of the prior trajectory of states leading to the current state $x^q$. ∎

In the case of MLS, the local search operator $\mathcal{L}$ transports the initial point $y$ through a sequence of neighborhood descents (always choosing the best point in the local neighborhood of the current solution) to some local solution $y'$. Thus, we may define an "entrapment" zone for each subset $L_j$ as follows:

$$A_j = \{x \in S \mid \mathcal{L}(x) \in L_j\},\ j = 1, \ldots, N.$$

(If $\mathcal{L}(x)$ is one-to-many, read $\mathcal{L}(x) \in L_j$ to mean one or more of the points from the mapping belong to $L_j$.) It is clear that $A_j$ ($L_j \subseteq A_j,\ \forall j$), has a non-zero volume in $S$ (under mild assumption, e.g. that $f$ is Lipscitz or Hölder). Divide $A_j$ into two mutually exclusive regions, $A'_j$ and $A''_j$, where $\mathcal{L}$ is one-to-one for all points in $A'_j$, and one-to-many for all points in $A''_j$. Then the probability that any iteration of MLS attains a local solution belonging to $L_j$ is given by

$$\gamma_j = P\{y \in A'_j\} + P\{y \in A''_j\}P\{\mathcal{L}(y) \in L_j \mid y \in A''_j\}.$$

Loosely-speaking, we see that $\gamma_j$ is proportional to the size of the entrapment zone $A_j$. Recalling that $S$ is a bounded region, it follows that $\gamma_j > 0, j = 1, \ldots, N$; furthermore, $\sum_{j=1}^{N} \gamma_j = 1$.

Let $Q_1$ = the number of iterations performed by MLS until a global solution is reached. It follows that $Q_1$ is a geometric random variable with parameter $\gamma_1$. Thus, the expected number of iterations required to obtain the global solution is $E[Q_1] = 1/\gamma_1$. We may similarly consider the number of iterations required to find a solution of value $f_r$ or better. This is again given by a geometric random variable $Q_r$ with parameter

$$\theta_r = \sum_{i=1}^{r} \gamma_i.$$

The expected value of $Q_r$, $E[Q_r] = \frac{1}{\theta_r}$.

The probability of obtaining a better solution in the subsequent iteration will be generally referred to as an "entrapment" probability. For MLS, and incumbent solution in $L_{r+1}$, this probability is given by $\theta_r$.

**Property 2** *The multistart local search heuristic (MLS) is globally convergent.*

*Proof.* Since $Q_1$ is a geometric random variable with parameter $\gamma_1 > 0$, the probability that the global solution is obtained in a finite number of iterations is equal to one. ∎

As seen above, the probability of obtaining a local solution $y' \in L_r$ in any iteration of MLS depends on the volume of the entrapment zone $A_r$ (or the number of points in $A_r$), since any point in $S$ is equally-likely to be chosen as the starting point of the iteration. Since the entrapment zone $A_1$ for the global solution tends to be an exponentially small proportion of $S$ as the problem size increases, the expected number of iterations $E[Q_1]$

tends to increase exponentially. Thus, although the procedure is globally convergent, the topology of the solution space implies an exponentially increasing time will be required to find a global solution. In practical terms, we must therefore concern ourselves with the finite time performance of the heuristic.

Now consider the set of neighborhood structures $\mathcal{N}_k$, $k = 1, \ldots, k_{max}$, used in the basic VNS procedure. Let

$$\mathcal{N}(x) = \bigcup_{k=1}^{k_{max}} \mathcal{N}_k(x)$$

denote the total neighborhood of any point $x \in S$.

**Definition 1** *$\mathcal{N}$ is said to* span *the solution space if $\mathcal{N}(x) = S \setminus \{x\}$, $\forall x \in S$.*

Note that this definition signifies that

$$\rho(x, y) \leq k_{max}, \forall x, y \in S.$$

Given that $\mathcal{N}$ spans the solution space, any point $y \in S$ may be reached from an incumbent solution $x$ in the *shaking* operation. This leads to the following rather obvious but fundamental result.

**Property 3** *If $\mathcal{N}$ spans $S$, the basic variable neighborhood search heuristic (VNS) is globally convergent.*

*Proof.* Consider a current incumbent solution $x$ with objective function value $f_r$, $r$ ¿ 1. Since $\mathcal{N}$ spans $S$, it is clear that a least one neighborhood $\mathcal{N}_k(x)$ may be found where the transition probability to a better solution is greater than zero. Thus, any incumbent solution $x$ that is not a global optimum must be a transient state. We conclude that the sequence of incumbents $\{x^q\}$ is absorbed in a finite number of iterations by a global optimum $(f(x^{N'}) = f_1)$. ∎

Given an incumbent solution $x$ with objective function value $f_{r+1}$, let $\sigma_{rk}$ = the probability that a better solution will be obtained from random starting point $y \in \mathcal{N}_k(x)$; i.e., $\sigma_{rk}$ is the entrapment probability associated with $\mathcal{N}_k(x)$. Then the probability of obtaining a better solution in the next iteration of VNS is given by

$$\sigma_r = \sigma_{r1} + \sum_{k=2}^{k_{max}} \sigma_{rk} \prod_{\ell=1}^{k-1} (1 - \sigma_{r\ell}).$$

We hypothesize in general that $\sigma_r \gg \theta_r$, the corresponding probability of improvement in the next iteration of MLS. Suppose as a simple numerical example that $S$ contains 1000 points, of which a total of 10 points are in entrapment zones that lead to an improved solution. Suppose further that there are 3 neighborhoods used, $\mathcal{N}_1(x), \mathcal{N}_2(x), \mathcal{N}_3(x)$, containing 9, 90 and 900 points, respectively; of the 10 *entrapment* points, one is found in

$\mathcal{N}_1(x)$, two in $\mathcal{N}_2(x)$, and the remaining seven in $\mathcal{N}_3(x)$. Then,

$$\sigma_r = \frac{1}{9} + \frac{2}{90}(1-\frac{1}{9}) + \frac{7}{900}(1-\frac{1}{9})(1-\frac{2}{90}) = 0.1377.$$

The expected number of equivalent MLS iterations (or local searches) for one iteration of VNS is obtained by the formula,

$$E[\#LS] = 1 \times \sigma_{r1} + \sum_{k=2}^{k_{max}-1} k\sigma_{rk} \prod_{\ell=1}^{k-1}(1-\sigma_{r\ell}) + k_{max} \prod_{\ell=1}^{k_{max}-1}(1-\sigma_{r\ell}),$$

for $k_{max} \geq 3$. For our example, this gives

$$E[\#LS] = 1 \times \frac{1}{9} + 2 \times \frac{2}{90}(1-\frac{1}{9}) + 3 \times (1-\frac{1}{9})(1-\frac{2}{90}) = 2.758.$$

Comparing $\sigma_r = 0.1377$ with $\theta_r \times E[\#LS] = 0.01 \times 2.758 = 0.02758$, we observe a five-fold increase in the probability of obtaining an improved solution resulting from the distribution of entrapment points within the "closer" neighborhoods of the incumbent solution.

The hypothesis that $\sigma_r \gg \theta_r$ is based on consistent empirical evidence that any local optimum tends to be located in close proximity to several other local optima in the solution space (e.g., see [1]). Thus, the probability of finding an improvement in a *tight* neighborhood of the incumbent solution is significantly larger than the probability of improvement from a randomly-selected starting point in $S$. Furthermore, the local search will be much faster in the *tight* neighborhood, since relatively few descent moves will be required.

**Property 4** *The finite time performance of basic VNS is statistically superior to MLS.*

An intuitive proof of Property 4 may proceed as follows. In the next iteration of MLS, the random starting point $y$ will most likely be chosen far away from the incumbent solution $x$. Since the furthest possible neighborhood predominates in size relative to the sum of the remaining neighborhoods, it is most probable that $\rho(x, y)$ is equal or close to the furthest possible distance (or largest possible $k_{max}$). If the probability $\theta_r$ of finding an improved solution is small, this implies that it is unlikely that $y$ will be one of the required entrapment points. Thus, we may expect several iterations of MLS before an improved solution is found. On the other hand, since the set of local solutions ($L$) tends to be found in a small compact subset of $S$ that contains $x$, we expect $\rho(x, x') <$ largest $k_{max}$, for any improved solution $x'$. Thus, the corresponding entrapment zones will tend to have dominant intersections with the closer neighborhoods centered at $x$. Since these neighborhoods are orders of magnitude smaller in size than the $k_{max}$ neighborhood, it follows that the probability of success in the next iteration of VNS is much higher. Empirical proof of this argument is obtained by comparing the finite time performance of the two methods: we expect to see a faster descent to better solutions using VNS. This fact will be demonstrated in the next section on sample instances of the continuous multisource Weber problem.

We conclude this section with a further note on convergence of VNS. Let

$$S_j = \cup_{t=1}^{j-1} A_t$$

denote the union of entrapment zones that may lead to better solutions than $f_j$,

$$\rho_j = \max_{x \in L_j}\{\min_{y \in S_j} \rho(x,y)\}, j = 1, \ldots, N-1,$$

and

$$k^*_{max} = \max_j\{\rho_j\}.$$

A sufficient condition for global convergence of VNS in combinatorial optimization problems is that $k_{max} \geq k^*_{max}$.

This provides a tighter requirement for convergence than the requirement that $N$ span the solution space (see Property 3). Thus, it is possible to choose smaller values of $k_{max}$ for more efficient versions of VNS, while still ensuring global convergence of the method.

# 4 Multisource Weber Problem

The location-allocation problem in the continuous plane can be formulated as follows ([11]):

$$\min f(U,V,Z) = \sum_{i=1}^{p} \sum_{j=1}^{n} z_{ij} \cdot w_j \cdot d_j(u_i, v_i)$$

subject to

$$\sum_{i=1}^{p} z_{ij} = 1, \quad j = 1, 2, \ldots, n,$$

$$z_{ij} \in [0,1] \quad i = 1, 2, \ldots, p; \; j = 1, 2, \ldots, n,$$

where $p$ facilities must be located to satisfy the demand of $n$ users, $(u_i, v_i)$ denotes the unknown location of the $i^{th}$ facility, $d_j(u_i, v_i)$ the Euclidean distance from the $i^{th}$ facility to the $j^{th}$ user, where the $j^{th}$ user is at given point $a_j = (a_{j1}, a_{j2})$, $w_j$ the demand (or weight) of the $j^{th}$ user, and $z_{ij}$ the fraction of this demand which is satisfied from the $i^{th}$ facility. It is well-known that an optimal solution exists with all $z_{ij} \in \{0,1\}$; i.e., each user is satisfied by its nearest facility with ties broken arbitrarily.

## 4.1 Finite time performance of MLS, VNS, and a hybrid method

Figure 1 gives the results for a problem with $n$=1060 customers taken from the TSP library ([14]). In this example we locate $p$=100 facilities. The MLS procedure uses a rapid search in a discretized $p$-median space. The effectiveness of this local search neighborhood is due to the asymptotic behavior of the discrete $p$-median and continuous multisource

Weber problems (e.g., see [8], [2]). We set the parameter $k_{max} = 10$ in VNS, and use *fast* $p$-median interchange neighborhoods ([15], [5]) for the shaking operation as explained in [2]. Note that $\mathcal{N}$ does not span the solution space with $k_{max} = 10$; so global convergence is not immediately guaranteed. On the other hand, the finite time performance is expected to improve for reasons discussed above.

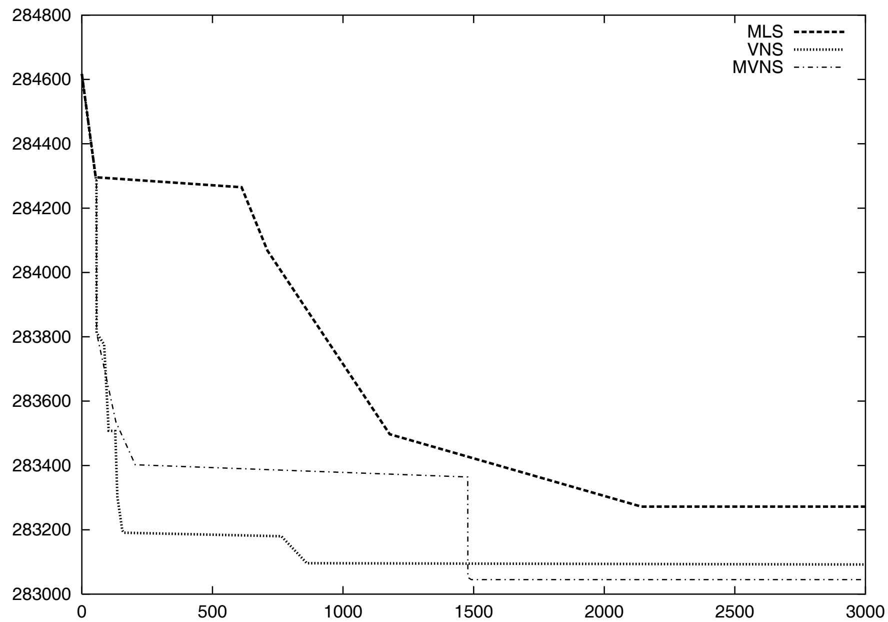


Figure 1: Comparison of finite time performance for $n = 1060$ and $p$=100.

A new method, multistart VNS (MVNS), is also examined. The idea is to combine the features of the two heuristics. If the VNS appears to get stuck in a large trough, the procedure restarts the VNS from a new randomly-selected point in $S$. In our implementation, the VNS is restarted each time an iteration through neighborhoods $\mathcal{N}_1(x), \ldots, \mathcal{N}_{k_{max}}(x)$ does not find an improvement to solution $x$.

The stopping criterion for the three methods is based on a maximum execution time, $t_{max} = 3,000$ seconds, as shown in Figure 1. The steeper initial descent (and larger number of descent moves) of VNS compared with MLS evidenced in Figure 1 is typical, and clearly demonstrates the superior performance of VNS. It is interesting to note that MVNS gives the best results of the three methods. In fact, the incumbent solution obtained by MVNS in only 1491 seconds and 23 re-starts of VNS is better than the best-known solution in the literature ([2]). Meanwhile the basic VNS generated many more unsuccessful iterations, almost 1000, through neighborhoods $\mathcal{N}_1, \ldots, \mathcal{N}_{k_{max}}$.

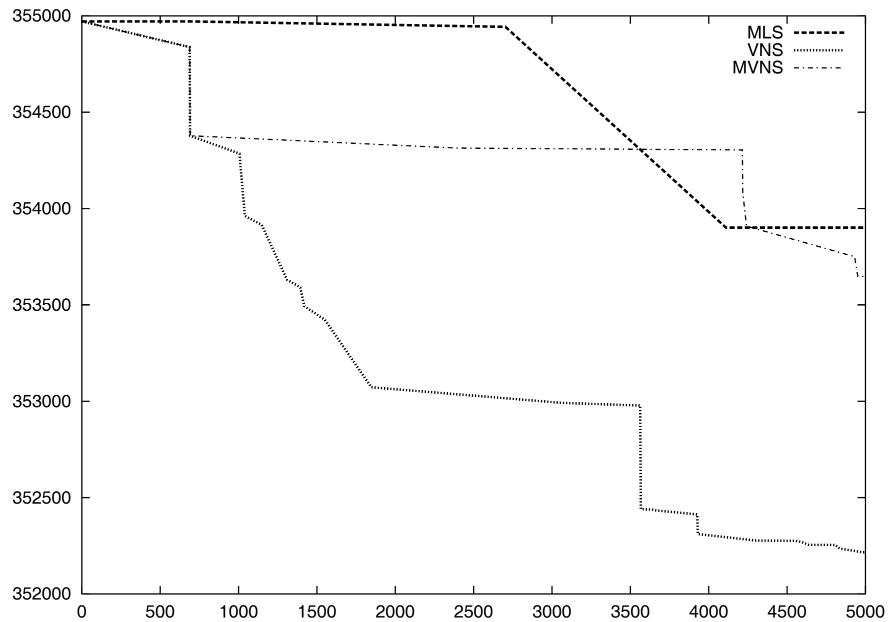


Figure 2: Comparison of finite time performance for $n = 3038$ and $p = 100$.

Figure 2 shows results for a larger problem ([14]) with $n = 3038$, $p = 100$. Here the basic VNS performs best. This is partly attributed to the fact that each restart in MLS and MVNS may use substantial CPU time descending from a poor solution.

## 4.2 Comparison of entrapment probabilities

We now examine the well-known 50-customer problem given in Eilon *et al.* [4]. In the first part of the experiment, 10,000 random restarts of Cooper's alternating (locate/allocate) heuristic [3], referred to as ALT for short, are performed for a specified value of $p$. The resulting local solutions are recorded, and sorted by objective function value. We are then able to estimate the probabilities associated with the entrapment zones:

$$\hat{\gamma}_j = \frac{q_j}{10,000},\ j = 1, \ldots, K,$$

where $q_j$ is the recorded number of occurrences of objective function value $f_j$ ( $\sum q_j = 10,000$), and $K$ is a lower bound on $N$. These point estimates may then be used to plot a cumulative probability curve as shown in Figure 3 for $p = 5, 10, 15$. Note that the horizontal axis measures the percent deviation from the optimal solution,

$$e_r = \frac{f_r - f^*}{f^*} \times 100,$$

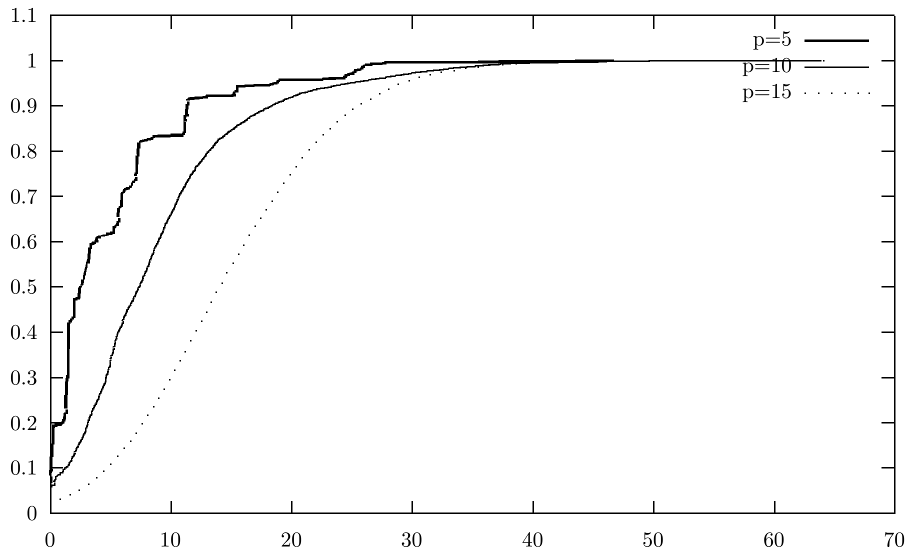


Figure 3: Cumulative probability versus % deviation from the optimal solution for the 50-customer problem.

where the optimal solution $f^*$ is obtained from [2]. The cumulative probability is calculated using the formula,

$$F(e_r) = \hat{\theta}_r = \sum_{j=1}^{r} \hat{\gamma}_j, \; r = 1, \ldots, K.$$

Referring to Figure 3, we observe that the cumulative probability curves shift to the right as $p$ increases. This signifies, as expected, that it is much harder for MLS to find good quality solutions as problem size increases, due to an exponentially-increasing number of local solutions. It is interesting to note that of the 10,000 iterations of MLS, the optimal solution was obtained 690, 34, 1 times for $p$=5, 10, 15, respectively, and the worst deviation from the optimal solution was, respectively, 46.74 %, 65.80 %, 70.27 %. In all 272, 3008, and 3363 different local solutions were obtained, respectively, for $p$ = 5, 10, 15.

In the next part of this experiment, we perform 10,000 iterations of VNS at a pre-selected local solution ($x$) found by MLS. The number of successes in each neighborhood of $x$ (i.e., the number of times a better solution is found from a randomly-chosen point in each neighborhood) is recorded, and used to estimate the entrapment probabilities ($\sigma_{rk}$), and from this, the net entrapment probability ($\sigma_r$) using the formula derived earlier, or simply as $\hat{\sigma}_r$ = total number of successes / 10,000.

Sample estimates obtained for $\theta_r$ and $\sigma_r$ are compared in Table 1. These results support the earlier claim that on average $\sigma_r \gg \theta_r$. We see in the table that $\sigma_r$ is in fact orders of magnitude larger than $\theta_r$ when the incumbent solution is "relatively" close in value to the optimal solution. This empirical evidence is consistent with the superior performance of VNS observed earlier on two larger problem instances.

| $j^{th}$ best solut. | $p = 5$ MLS $(\theta_{j-1})$ | $p = 5$ VNS $(\sigma_{j-1})$ | $p = 10$ MLS $(\theta_{j-1})$ | $p = 10$ VNS $(\sigma_{j-1})$ | $p = 15$ MLS $(\theta_{j-1})$ | $p = 15$ VNS $(\sigma_{j-1})$ |
|---|---|---|---|---|---|---|
| 2 | 0.0690 | 0.3364 | 0.0034 | 0.2636 | 0.0001 | 0.3192 |
| 3 | 0.0875 | 0.6664 | 0.0074 | 0.6112 | 0.0002 | 0.3851 |
| 4 | 0.1002 | 0.7248 | 0.0092 | 0.6972 | 0.0003 | 0.2291 |
| 5 | 0.1019 | 0.4632 | 0.0104 | 0.0578 | 0.0005 | 0.2513 |
| 6 | 0.1504 | 0.8981 | 0.0139 | 0.6776 | 0.0006 | 0.2995 |
| 7 | 0.1805 | 0.5267 | 0.0201 | 0.4582 | 0.0007 | 0.3310 |
| 8 | 0.1879 | 0.6060 | 0.0210 | 0.5508 | 0.0008 | 0.3992 |
| 9 | 0.1903 | 0.7928 | 0.0239 | 0.8784 | 0.0009 | 0.4893 |
| 10 | 0.1916 | 0.9403 | 0.0241 | 0.9674 | 0.0010 | 0.5226 |
| 11 | 0.1991 | 0.6584 | 0.0242 | 0.3177 | 0.0012 | 0.4891 |
| 12 | 0.2068 | 0.2362 | 0.0269 | 0.6912 | 0.0013 | 0.5752 |
| 13 | 0.2106 | 0.7669 | 0.0273 | 0.6547 | 0.0014 | 0.1420 |
| 14 | 0.2129 | 0.5777 | 0.0299 | 0.5754 | 0.0016 | 0.2788 |
| 15 | 0.2189 | 0.9915 | 0.0314 | 0.6441 | 0.0017 | 0.6283 |
| 16 | 0.2713 | 0.9228 | 0.0340 | 0.3922 | 0.0019 | 0.2604 |
| 17 | 0.2861 | 0.1875 | 0.0357 | 0.4832 | 0.0021 | 0.5176 |
| 18 | 0.4168 | 0.8619 | 0.0388 | 0.6268 | 0.0022 | 0.4238 |
| 19 | 0.4264 | 0.8870 | 0.0398 | 0.8595 | 0.0023 | 0.3297 |
| 20 | 0.4280 | 0.8899 | 0.0405 | 0.8056 | 0.0025 | 0.4131 |

Table 1: Entrapment probabilities of MLS and VNS.

# 5 Conclusions

The global convergence properties of multistart local search (MLS) and variable neighborhood search (VNS) are examined under general conditions. The finite time performance of VNS is argued to be statistically better than MLS. Two large instances of the continuous location-allocation problem demonstrate this result. In addition, a multistart variable neighborhood search (MVNS) is introduced that combines the features of MLS and VNS. The concept of "entrapment" probabilities is introduced to assist in the analysis.

Much theoretical and empirical work remains to be done to improve our understanding and implementation of these and other metaheuristics.

## References


[1] K.D. Boese, A.B. Kahng and S. Muddu. A New Adaptive Multi-Start Technique for Combinatorial Global Optimizations. *Oper. Res. Lett.*, 16: 101–113, 1994.

[2] J. Brimberg, P. Hansen, N. Mladenović and É. Taillard. Improvements and Comparison of Heuristics for Solving the Multisource Weber Problem. *Oper. Res.*, 48(3): 444–460, 2000.

[3] L. Cooper. Heuristic Methods for Location-Allocation Problems. *SIAM Rev.*, 6: 37–53, 1964.

[4] S. Eilon, C.D.T. Watson-Gandy, N. Christofides. *Distribution Management*, Hafner, New York, 1971.

[5] P. Hansen and N. Mladenović. Variable Neighborhood Search for the $p$-Median, *Location Sci.*, 5: 207–226, 1997.

[6] P. Hansen and N. Mladenović. Variable Neighborhood Search: Principles and Applications. *European J. Oper. Res.* 130(3): 449–467, 2001.

[7] P. Hansen and N. Mladenović. Developments of Variable Neighborhood Search, C. Ribeiro, P. Hansen (eds.), *Essays and Surveys in Metaheuristics*, pp. 415–439, Kluwer Academic Publishers, Boston/Dordrecht/London, 2001.

[8] P. Hansen, N. Mladenović and É. Taillard. Heuristic Solution of the Multisource Weber Problem as a $p$-Median Problem, *Oper. Res. Lett.*, 22: 55–62, 1998.

[9] S. Jacobson. Analyzing the Performance of Local Search Algorithms Using Generalized Hill Climbing Algorithms, C. Ribeiro, P. Hansen (eds), *Essays and surveys in metaheuristics*, pp. 441–467, Kluwer Academic Publishers, Boston/Dordrecht/London, 2001.

[10] A.W. Jonson and S.H. Jacobsen. On the Convergence of Generalized Hill Climbing Algorithms, to appear in *Discr. App. Math.*

[11] R.E. Kuenne and R.M. Soland. Exact and Approximate Solutions to the Multisource Weber Problem. *Math. Programming*, 3: 193–209, 1972.

[12] N. Mladenović and P. Hansen. Variable Neighborhood Search. *Computers Oper. Res.* 24: 1097–1100, 1997.

[13] H. Mulenbein and J. Zimmermann. Size of Neighborhood More Important than Temperature for Stochastic Local Search, Tech. report, GMD Labaratory, Germany, 2002.

[14] G. Reinelt. TSLIB-A Traveling Salesman Library. *ORSA J. on Computing*, 3: 376–384, 1991.

[15] R. Whitaker. A Fast Algorithm for the Greedy Interchange for Large-Scale Clustering and Median Location Problems. *INFOR*, 21: 95–108, 1983.